TEXT CLASSIFICATION AND CLASSIFIERS: A SURVEY

Vandana Korde

Sardar Vallabhbhai National Institute of Technology, Surat

korde.vandana@gmail.com

C Namrata Mahender

Department of Computer Science&IT, Dr.B.A.M.U Aurangabad

cc_namrata@yahoo.co.in

Abstract

As most information (over 80%) is stored as text, text mining is believed to have a high commercial potential value. knowledge may be discovered from many sources of information; yet, unstructured texts remain the largest readily available source of knowledge .Text classification which classifies the documents according to predefined categories .In this paper we are tried to give the introduction of text classification, process of text classification as well as the overview of the classifiers and tried to compare the some existing classifier on basis of few criteria like time complexity, principal and performance.

Keywords

Text classification, Text Representation, Classifiers

1. INTRODUCTION

The text mining studies are gaining more importance recently because of the availability of the increasing number of the electronic documents from a variety of sources. Which include unstructured and semi structured information. The main goal of text mining is to enable users to extract information from textual resources and deals with the operations like, retrieval, classification (supervised, unsupervised and semi supervised) and summarizationNatural Language Processing (NLP), Data Mining, and Machine Learning techniques work together to automatically classify and discover patterns from the different types of the documents [1].

Text classification (TC) is an important part of text mining, looked to be that of manually building automatic TC systems by means of knowledge-engineering techniques, i.e. manually defining a set of logical rules that convert expert knowledge on how to classify documents under the given set of categories. For example would be to automatically label each incoming news story with a topic like “sports”, “politics”, or “art”. a data mining classification task starts with a training set D = (d_1….. d_n) of documents that are already labelled with a class C1,C2 (e.g. sport, politics). The task is then to determine a classification model which is able to assign the correct class to a new document d of the domain Text classification has two flavours as single label and multi-label .single label document is belongs to only one class and multi label document may be belong to more than one classes In this paper we are consider only single label document classification.

DOI : 10.5121/ijaia.2012.3208

The remainder of the paper is organized as follows .Section 2 given the process of Text classification out of these we will more consternate on classification stage ,will see the detail of classifier (KNN, NB, SVM, LLSF, Centroid and Associative etc) which is in Section 3 . In Section 4 Comparative observation is given and finally in Section 5 conclusion were made

2. TEXT CLASSIFICATION PROCESS

The stages of TC are discussing as following points.

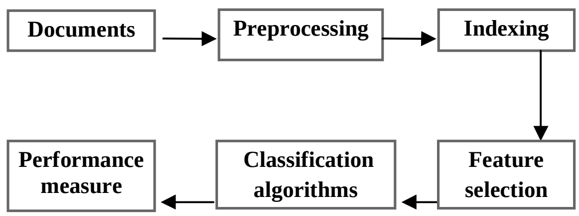

Fig. 1 Document Classification Process

2.1 Documents Collection

This is first step of classification process in which we are collecting the different types (format) of document like html, .pdf, .doc, web content etc.

2.2 Pre-Processing

The first step of pre-processing which is used to presents the text documents into clear word format. The documents prepared for next step in text classification are represented by a great amount of features. Commonly the steps taken are:

Tokenization: A document is treated as a string, and then partitioned into a list of tokens. *Removing stop words*: Stop words such as “the”, “a”, “and”, etc are frequently occurring, so the insignificant words need to be removed.

Stemming word: Applying the stemming algorithm that converts different word form into similar canonical form. This step is the process of conflating tokens to their root form, e.g. connection to connect, computing to compute

2.3 Indexing

The documents representation is one of the pre-processing technique that is used to reduce the complexity of the documents and make them easier to handle, the document have to be transformed from the full text version to a document vector The Perhaps most commonly used document representation is called vector space model (SMART) [55] vector space model, documents are represented by vectors of words. Usually, one has a collection of documents which is represented by word by word document Matrix. BoW/VSM representation scheme has its own limitations. Some of them are: high dimensionality of the representation, loss of correlation with adjacent words and loss of semantic relationship that exist among the terms in a document.to To overcome these problems, term weighting methods are used to assign appropriate weights to the term as shown in following matrix

$$\left(\begin{array}{cccccc}
 & T_1 & T_2 & \dots & T_{at} & C_i \\
D_1 & w_{11} & w21 & \dots & w_{t1} & c_1 \\
D_2 & w_{12} & w22 & \dots & w_{t2} & c_2 \\
: & : & : & & : & \\
Dn & w_{1n} & w2n & \dots & w_{tn} & C_n
\end{array}\right)$$

Where each entry represents the occurrence of the word in the document, where w_{tn} is the weight of word i in the document n .since every word does not normally appear in each document, .There are several way of determining the weight w_{11}. Like Boolean weighting, word frequency weighting, tf-idf, entropy etc. But the major drawback of this model is that it results in a huge sparse matrix, which raises a problem of high dimensionality. Other various methods are presented in [56] as 1) an ontology representation for a document to keep the semantic relationship between the terms in a document.2) a sequence of symbols (byte, a character or a word) called N-Grams, that are extracted from a long string in a document., it is very difficult to decide the number of grams to be considered for effective document representation.3) multi-word terms as vector components .But this method requires a sophisticated automatic term extraction algorithms to extract the terms automatically from a document 4) Latent Semantic Indexing (LSI) which preserves the representative features for a document, The LSI preserves the most representative features rather than discriminating features. Thus to overcome this problem 5) Locality Preserving Indexing (LPI), discovers the local semantic structure of a document. But is not efficient in time and memory 6) a new representation to model the web documents is proposed. HTML tags are used to build the web document representation.

2.4 Feature Selection

After pre-processing and indexing the important step of text classification, is feature selection [2] to construct vector space, which improves the scalability, efficiency and accuracy of a text classifier. The main idea of Feature Selection (FS) is to select subset of features from the original documents. FS is performed by keeping the words with highest score according to predetermined measure of the importance of the word. Because of for text classification a major problem is the high dimensionality of the feature space. Many feature evaluation metrics have been notable among which are information gain (IG), term frequency, Chi-square, expected cross entropy, Odds Ratio, the weight of evidence of text, mutual information, Gini index. But FS of association word mining is more efficient than IG and document frequency [57] .Other various methods are presented like [58] sampling method which is randomly samples roughly features and then make matrix for classification. By considering problem of high dimensional problem [59] is presented new FS witch use the genetic algorithm (GA) optimization.

2.5 Classification

The automatic classification of documents into predefined categories has observed as an active attention, the documents can be classified by three ways, unsupervised, supervised and semi supervised methods. From last few years, the task of automatic text classification have been extensively studied and rapid progress seems in this area, including the machine learning approaches such as Bayesian classifier, Decision Tree, K-nearest neighbor(KNN), Support Vector Machines(SVMs), Neural Networks, Rocchio's. Some techniques are described in section 3.

2.6 Performance Evaluations

This is Last stage of Text classification, in which the evaluations of text classifiers is typically conducted experimentally, rather than analytically. The experimental evaluation of classifiers, rather than concentrating on issues of Efficiency, usually tries to evaluate the effectiveness of a classifier, i.e. its capability of taking the right categorization decisions. An important issue of Text categorization is how to measures the performance of the classifiers. Many measures have been used, like Precision and recall [54]; fallout, error, accuracy etc. are given below

Precision wrt ci (Pri) is defined as the as the probability that if a random document *dx* is classified under *ci*, this decision is correct. Analogously, *Recall wrt ci* (*Rei*) is defined as the conditional that, if a random document *dx* ought to be classified under *ci*, this decision is taken

TP_i–The number of document correctly assigned to this category.

FN - The number of document incorrectly assigned to this category

FPi - The number of document incorrectly rejected assigned to this category

TNi - The number of document correctly rejected assigned to this category

Fallout = FNi / FNi + TNi

Error = FNi +FPi / TP_i + FNi +FPi +TNi

Accuracy = TP_i + TNi

For obtaining estimates of precision and recall relative to the whole category set, two different methods may be adopted Micro-averaging and Macro-averaging some other measures are also use as Break–even point, F-measure, Interpolation [55] . In next section we will continue with Classifiers.

3. CLASSIFIER

3.1 Rocchio's Algorithm

Rocchio's learning algorithm [6] is in the classical IR tradition. It was originally designed to use relevance feedback in querying full-text databases, Rocchio's Algorithm is a vector space method for document routing or filtering in informational retrieval, build prototype vector for each class using a training set of documents, i.e. the average vector over all training document vectors that belong to class c_i, and calculate similarity between test document and each of prototype vectors, which assign test document to the class with maximum similarity.

Ci= α * centroid ci − β * centroid ~ci.[7] gives find similar method as of Rocchio is use in inductive learning process to find similarity between test example and category centroid using all feature .This algorithm is easy to implement, efficient in computation. The researchers have used a variation of Rocchio's algorithm in a machine learning context,[8].

3.2 K-Nearest Neighbors

K-NN classifier is a case-based learning [9] algorithm that is based on a distance or similarity function for pairs of observations, such as the Euclidean distance or Cosine similarity measure's This method is try for many application [10] Because of its effectiveness, non-parametric and easy to implementation properties, however the classification time is long and difficult to find optimal value of k .The best choice of k depends upon the data; generally, larger values of k reduce the effect of noise on the classification, but make boundaries between classes less distinct. A good k can be selected by various heuristic techniques .to overcome this drawback [11] modify

traditional KNN with different K-values for different classes rather than fixed value for all classes Fang Lu have been try to improve performance of KNN by using WKNN [12].
A major drawback of the similarity measure used in k-NN is that it uses all features in computing distances. In many document data sets, only smaller number of the total vocabulary may be useful in categorizing documents. A possible approach to overcome this problem is to learn weights for different features (or words in document data etc). [12] Propose the Weight Adjusted k-Nearest Neighbor (WAKNN) classification algorithm that is based on the k-NN classification paradigm. With the help of KNN can improve the performance of text classification [13] from training set and also accuracy can improve with combination of KNN [14] with another method

3.3 Naïve Bayes

Naïve bias method is kind of module classifier [15] under known priori probability and class conditional probability .it is basic idea is to calculate the probability that document D is belongs to class C. There are two event model are present for naive Bias [16] [17] [18] as multivariate Bernoulli and multinomial model. Out of these model multinomial model is more suitable when database is large, but there are identifies two serious problem with multinomial model first it is rough parameter estimated and problem it lies in handling rare categories that contain only few training documents. They [19] propose Poisson model for NB text classification and also give weight enhancing method to improve the performance of rare categories. Modified NB is propose [20] to improve performance of text classification, also [21] provides ways to improve naive Bayes classification by searching the dependencies among attribute. Naïve Bayes is easy for implementation and computation. So it is use for pre-processing [22] i.e. For vectorization. Performance of naïve bias is very poor when features are highly correlated and, highly it is sensitive to feature selection so the [23] propose two metrics for NB which applied on multiclass text document.

3.4 Decision tree

When decision tree is used for text classification it consist tree internal node are label by term, branches departing from them are labeled by test on the weight, and leaf node are represent corresponding class labels .Tree can classify the document by running through the query structure from root to until it reaches a certain leaf, which represents the goal for the classification of the document. Most of training data will not fit in memory decision tree construction it becomes inefficient due to swapping of training tuples. To handle this issue [24] presents method which can handle numeric and categorical data.

New method is proposing [25] as FDT to handle the multi-label document witch reduce cost of induction, and [26] presented decision-tree-based symbolic rule induction system for text categorization which also improves text classification. The decision tree classification method is outstanding from other decision support [27] tools with several advantages like its simplicity in understanding and interpreting, even for non-expert users. So for that it is used in some application [28]

3.5 Decision Rule

Decision rules classification method uses the rule-based inference to classify documents to their annotated categories [29]. A popular format for interpretable solutions is the disjunctive normal form (DNF) model. [30] A classifier for category ci built by an inductive rule learning method consists of a disjunctive normal form (DNF) rule. [4]. In the case of handling a dataset with large number of features for each category, heuristics implementation is recommended to reduce the size of rules set without affecting the performance of the classification The [31] presents a hybrid method of rule based processing and back-propagation neural networks for spam filtering.

3.6 SVM

The application of Support vector machine (SVM) method to Text Classification has been propose by [32]. The SVM need both positive and negative training set which are uncommon for other classification methods. These positive and negative training set are needed for the SVM to seek for the decision surface that best separates the positive from the negative data in the n dimensional space, so called the hyper plane. The document representatives which are closest to the decision surface are called the support vector.

SVM classifier method is outstanding from other with its effectiveness [5] to improve performance of text classification [34] combining the HMM and SVM where HMMs are used to as a feature extractor and then a new feature vector is normalized as the input of SVMs, so the trained SVMs can classify unknown texts successfully, also by combing with Bayes [33] use to reduce number of feature which as reducing number of dimension .SVM is more capable [35] to solve the multi-label class classification

3.7 Neural Network

A neural network classifier is a network of units, where the input units usually represent terms, the output unit(s) represents the category. For classifying a test document, its term weights are assigned to the input units; the activation of these units is propagated forward through the network, and the value that the output unit(s) takes up as a consequence determines the categorization decision. Some of the researches use the single-layer perceptron, due to its simplicity of implementing [36]. The multi-layer perceptron which is more sophisticated, also widely implemented for classification tasks [37].Models using back-propagation neural network (BPNN) and modified back-propagation neural network (MBPNN) are proposed in [38] for documents classification. An efficient feature selection method [39] is used to reduce the dimensionality as well as improve the performance. New Neural network based document classification method. [40] Was presented, which is helpful for companies to manage patent documents more effectively

.

3.8 LLSF

LLSF stands for Linear Least Squares Fit, a mapping approach developed by Yang [41]. The training data are represented in the form of input/output vector pairs where the input vector is a document in the conventional vector space model (consisting of words with weights), and output vector consists of categories (with binary weights) of the corresponding document. Basically this method is used for Information Retrieval [42] for giving the output of query in form of relevant document but it can easily use for text classification. LLSF is one of the most effective text classifiers known to date. One of its disadvantages, though, is that the computational cost of computing the matrix is much higher than that of many other competitors in the TC arena

3.9 Voting

This algorithm is based on method of classifier committees and is based on idea that given task that requires expert opinion knowledge to be performed. k experts opinion may be better than one if their individual judgments are appropriately combined. Different combination rules are present as the simplest possible rule is majority voting (MV)If two or three classifiers are agree on a class for a test document, the result of voting classifier is that class. Second weighted majority voting, in this method, the weights are specific for each class in this weighting method, error of each classifier is calculated. Other two rule are presented by [43] as DCS (dynamic classifier selection) whereby among committee {K1... Kn} the classifier Kt that yields the best effectiveness on the l validation examples most similar to dj is selected, and its judgment adopted by the committee. Still different policy, somehow intermediate between WLC and DCS, is adaptive classifier combination (ACC), whereby the judgments of all the classifiers in the committee are summed together, but their individual contribution is weighted by the effectiveness. [43] [44] has used combinations of different classifiers with different functions. , This method is easy to implement and understand but it takes long time for giving result.

3.10 Associative classifier

Recent studies in the data mining community proposed new methods for classification employing association rule mining. These associative classifiers have proven to be powerful and achieve high accuracy. [45]. The main idea behind this algorithm is to scan the transactional database searching for k-item sets relationships among items in a transactional database To Build an Associative Text Classifier construction phases are shown in following figure.

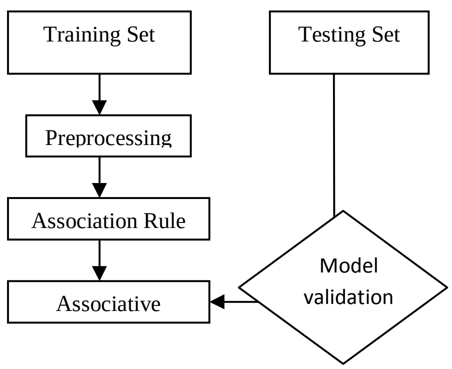

Fig. 2: Construction phases for an association-rule-based text categorizer

The first three steps belong to the training process while the last one represents the testing (or classification) phase. More details on the process are given in the subsections below Collecting the training set document after performing the pre-processing; in second phase using association algorithm on the documents would generate a very large number of association rules. There are some issues as huge amount of rules contain noisy information which would mislead the classification process, another is would make the classification time longer. So pruning is required in which the set of rules that were selected after pruning phase represents actual classifier.

The classification process searches in this set of rules for finding those classes that are closest to be attached with the documents present for categorization. [46] Introduce new algorithm for text classification association rule base text classifier, instead of taking simple join ARTC join (any two item in an item set can be joined if they have same category). Some researcher [47] are used the association mining to discover the set of association word in documents that acts as features, then classify a new document using derived feature set same as [48] use this association rule with decision tree which gives better performance than classic algorithm.

3.11 Centroid based classifier

The centroid-based classification algorithm is very simple. [50] [51] For each set of documents belonging to the same class, we compute their centroid vectors. If there are k classes in the training set, this leads to k centroid vectors (C_1, C2, C3...) where each Cn is the centroid for the jet class. The class of a new document x is determined as, First the document-frequencies of the various terms computed from the training set Then, compute the similarity between x to all k centroid using the cosine measure. Finally, based on these similarities, and assign x to the class corresponding to the most similar centroid

3.12 Additional classifier

In the previous sections we have tried to give an overview as complete as possible of the approaches that have been proposed in TC. Although for reasons of space we will not discuss them in detail, we at least want to mention the existence of WORD, Sleeping expert, CONSTRUE [54]**,** genetic, Online Classifie**r** [4] Fuzzy correlation and some Hybrid technique are given in [3].

4. COMPARATIVE OBSERVATIONS

The performance of a classification algorithm is greatly affected by the quality of data source. Irrelevant and redundant features of data not only increase the cost of mining process, but also reduce the Quality of the result in some cases [3]. Each algorithm has its own advantages and disadvantages as described in Table.1 with their time complexity by taking considering summary from [49][52] .The works in [5] [54] compare the most common method in most cases support machine and K-nearest neighbor have better effect neural network is after then and then naïve bays is last and its evaluation index is again break –even point

5. CONCLUSIONS

The growing use of the textual data witch needs text mining, machine learning and natural language processing techniques and methodologies to organize and extract pattern and knowledge from the documents. This review focused on the existing literature and explored the documents representation and an analysis of feature selection methods and classification algorithms were presented. It was verified from the study that information Gain and Chi square statistics are the most commonly used and well performed methods for feature selection, however many other FS methods are proposed. This paper also gives a brief introduction to the various text representation schemes. The existing classification methods are compared and contrasted based on various parameters namely criteria used for classification, algorithms adopted and classification time complexities. From the above discussion it is understood that no single representation scheme and classifier can be mentioned as a general model for any application. Different algorithms perform differently depending on data collection. However, to the certain extent SVM with term weighted VSM representation scheme performs well in many text classification tasks.

6. REFERENCES

[1] F. Sebastiani, “Text categorization”, Alessandro Zanasi (ed.) Text Mining and its Applications, WIT Press, Southampton, UK, pp. 109-129, 2005.

[2] A. Dasgupta, P. Drineas, B. Harb, “Feature Selection Methods for Text Classification”, KDD’07, ACM, 2007.

[3] A. Khan, B. Baharudin, L. H. Lee, K. Khan, “A Review of Machine Learning Algorithms for Text-Documents Classification”, Journal of Advances Information Technology, vol. 1, 2010.

[4] F. Sebastiani, “Machine Learning in Automated Text Categorization”, ACM 2002.

[5] Y. Y. X. Liu, “A re-examination of Text categorization Methods” IGIR-99, 1999.

[6] Hein Ragas Cornelis H.A. Koster, “Four text classification algorithms compared on a Dutch corpus” SIGIR 1998: 369-370 1998.

[7] Susan Dumais John Platt David Heckerman, “Inductive Learning Algorithms and Representations for Text Categorization”, Published by ACM, 1998.

[8] Michael Pazzani Daniel Billsus “Learning and Revising User Profiles: The Identification of Interesting Web Sites”, Machine Learning, 313–331 1997

[9] Gongde Guo, Hui Wang, David Bell, Yaxin Bi and Kieran Greer, “KNN Model-Based Approach in Classification”, Proc. ODBASE pp- 986 – 996, 2003

[10] Eiji Aramaki and Kengo Miyo, “Patient status classification by using rule based sentence extraction and bm25-knn based classifier”, Proc. of i2b2 AMIA workshop, 2006.

[11] Muhammed Miah, “Improved k-NN Algorithm for Text Classification”, Department of Computer Science and Engineering University of Texas at Arlington, TX, USA.

[12] Fang Lu Qingyuan Bai, “A Refined Weighted K-Nearest Neighbours Algorithm for Text Categorization”, IEEE 2010.

[13] Kwangcheol Shin, Ajith Abraham, and Sang Yong Han, “Improving kNN Text Categorization by Removing Outliers from Training Set”, Springer-Verlag Berlin Heidelberg 2006.

[14] Methods Ali Danesh Behzad Moshiri “Improve text classification accuracy based on classifier fusion methods”. 10th International Conference on Information Fusion, 1-6 2007.

[15] SHI Yong-feng, ZHAO, “Comparison of text categorization algorithm”, Wuhan university Journal of natural sciences. 2004.

[16] D. Lewis, “Naive Bayes at Forty: The Independence Assumption in Information Retrieval”, Proc. ECML-98, 10th European Conf. Machine 1998.

[17] Vidhya. K.A G.Aghila, “A Survey of Naïve Bayes Machine Learning approach in Text Document Classification”, (IJCSIS) International Journal of Computer Science and Information Security, Vol. 7, 2010.

[18] McCallum, A. and Nigam K., "A Comparison of Event Models for Naive Bayes Text Classification". AAAI/ ICML -98 Workshop on Learning for Text Categorization

[19] Sang- Bum Kim, et al, “Some Effective Techniques for Naive Bayes Text Classification “IEEE Transactions on Knowledge and Data Engineering, Vol. 18, November 2006.

[20] Yirong Shen and Jing Jiang” Improving the Performance of Naive Bayes for Text Classification”CS224N Spring 2003

[21] Michael J. Pazzani “Searching for dependencies in Bayesian classifiers” Proceedings of the Fifth Int. workshop on AI and, Statistics. Pearl, 1988

[22] Dino Isa “Text Document Pre-Processing Using the Bayes Formula for Classification Based on the Vector Space Mode”, Computer and Information Science November, 2008

[23] Bayes Jingnian Chen a, b, Houkuan Huang a, Shengfeng Tian a, Youli Qua a “Feature selection for text classification with Naïve”, China Expert Systems with Applications 36 5432–54352009

[24] Mnish Mehta, Rakesh agrwal” SLIQ: A Fast Scalable Classifier for Data Mining” 1996.

[25] Peerapon Vateekul and Miroslav Kubat, “Fast Induction of Multiple Decision Trees in Text Categorization From Large Scale,Imbalanced, and Multi-label Data”, IEEE International Conference on Data MiningWorkshops 2009

[26] D. E. Johnson F. J. Oles T. Zhang T. Goetz, “A decision-tree-based symbolic rule induction system for text Categorization”, by IBM SYSTEMS JOURNAL, VOL 41, NO 3, 2002

[27]David D. Lewis and Marc Ringuette, “A comparison of two learning algorithms for text categorization”, Proceedings of SDAIR-94, 3rd Annual Symposium on Document Analysis and Information Retrieval, Las Vegas, US 1994.
[28] HAO CHEN, YAN ZHAN, YAN LI, “The Application Of Decision Tree In Chinese Email Classification”, Proceedings of the Ninth International Conference on Machine Learning and Cybernetics, Qingdao, 11-14 July 2010
[29] C.Apte, F. Damerau, and S.M. Weiss “Automated Learning of Decision Rules for Text Categorization”, ACM Transactions on Information Systems, 1994
[30] Sholom M. Weiss Nitin Indurkhya, “Rule-based Machine Learning Methods for Functional Prediction”, Journal of Artificial Intelligence Research 3 383-403 1995
[31] Chih-Hung Wu “Behavior-based spam detection using a hybrid method of rule-based Techniques and neural networks”, Expert Systems with Applications 36 4321– 4330 2009
[32] Joachims, T. “Text categorization with support vector machines: learning with many relevant features”. In Proceedings of ECML-98, 10th European Conference on Machine Learning (Chemnitz, DE), pp. 137–142 1998.
[33] Loubes, J. M. and van de Geer, S “Support vector machines and the Bayes rule in classification”, Data mining knowledge and discovery 6 259-275.2002
[34] Chen donghui Liu zhijing, “A new text categorization method based on HMM and SVM”, IEEE2010
[35] Yu-ping Qin Xiu-kun Wang, “Study on Multi-label Text Classification Based on SVM” Sixth International Conference on Fuzzy Systems and Knowledge Discovery 2009
[36] Dagan, I., Karov, Y., and Roth, D. “Mistake-Driven Learning in Text Categorization.” In Proceedings of CoRR. 1997
[37] MIgual E .Ruiz, Padmini Srinivasn, “Automatic Text Categorization Using Neural networks”, Advaces in Classification Research, Volume VIII.
[38] Cheng Hua Li , Soon Choel Park “An efficient document classification model using an improved back propagation neural network and singular value decomposition”, Expert Systems with Applications, 3208–3215, 2009
[39] Hwee TOU Ng Wei Boon Goh Kok Leong Low, “Feature Selection, Perception Learning, and a Usability Case Study for Text Categorization”, SIGIR 97 Philadelphia PA,
[40] Amy J.C. Trappey a, Fu-Chiang Hsu a, Charles V. Trappey b, Chia-I. Lin “Development of a patent document classification and search platform using a back-propagation network”, Expert Systems with Applications 31 755–765 2006
[41] Yiming Yang And Christopher G. Chute Mayo Cllnic “An Example-Based Mapping Method For Text Categorization And Retrieval” ACM Transactions On Information Systems, Vol. 12, No 3, Pages 252-277, July 1994
[42] Yiming Yang Christopher G. Chute “A Linear Least Squares Fit Mapping Method For Information Retrieval From Natural Language Texts” Acres De Coling-92 Nantes, 23-28 AOUT 1992
[43] Li, Y. H. and Jain, A. K. “Classification of text documents”. The Computer Journal, 537–546. 1998.
[44] Larkey, L. S. and Croft, W. B. “Combining classifiers in text categorization”. In Proceedings of SIGIR-96, 19th ACM International Conference on Research and Development in Information Retrieval (Zurich, CH, 1996), pp. 289–297 1996
[45] O. Zaiane, and M. Antonie, “Text Document Categorization by Term Associaton”, Proceedings of ICDM 2002, IEEE, , pp.19-26 2002
[46] Supaporn Buddeewong1 and Worapoj Kreesuradej” A New Association Rule-Based Text Classifier Algorithm”, Proceedings of the 17th IEEE International Conference on Tools with Artificial Intelligence, 2005
[47] S. M. Kamruzzaman, Chowdhury Mofizur Rahman: “Text Categorization using Association Rule and Naive Bayes Classifier” CoRR, 2010
[48] Mohammad Masud Hasan and Chowdhury Mofizur Rahman,” Text Categorization Using Association Rule Based Decision Tree”, Proceeding of the 6th International Conference on Computer and Information Technology (ICCIT), pp 453-456, Bangladesh, 2003
[49] Sholom M. Weiss, Chidanand Apte, Fred J. Damerau, David E. Johnson, Frank J. Oles, Thilo Goetz, and Thomas Hampp, IBM T.J. Watson Research Center “Maximizing Text-Mining Performance” 1094-7167/99 IEEE INTELLIGENT SYSTEMS. 1999
[50] Songbo Tan ”An improved centroid classifier for text categorization”
Expert Systems with Applications xxx 2007

[51] Eui-Hong (Sam) Han and George Karypis "Centroid-Based Document Classification: Analysis & Experimental Results" PKDD '00 Proceedings of the 4th European Conference on Principles of Data Mining and Knowledge Discovery Springer-Verlag London, UK ©2000.
[52] B S Harish, D S Guru, S Manjunath " Representation and Classification of Text Documents: A Brief Review" IJCA Special Issue on "Recent Trends in Image Processing and Pattern Recognition"RTIPPR, 2010.
[53] Shi Yong-Feng, Zhao Yan-Ping in Wuhan " Comparison of Text Categorization Algorithms " University Journal of Natural Sciences 2004
[54] Yiming Yang "An Evolution of statistical Approaches to Text Categorization" Information Retrieval 1, 69-90 1999.
[55] Kjersti Aas and Line Eikvil "Text Categorization: A Survey" Report No. 941. ISBN 82-539-0425-8. , June, 1999.
[56] B S Harish, D S Guru, S Manjunath "Representation and Classification of Text Documents: A Brief Review" IJCA Special Issue on "Recent Trends in Image Processing and Pattern Recognition" RTIPPR, 2010.
[57] Su-Jeong Ko and Jung-Hyun Lee "Feature Selection Using Association Word Mining for Classification "H.C. Mayr et al. (Eds.): DEXA 2001, LNCS 2113, pp. 211–220, 2001.
[58] Anirban Dasgupta "Feature Selection Methods for Text Classification "KDD'07, August 12–15, 2007.
[59] Wei Zhao "A New Feature Selection Algorithm in Text Categorization "International Symposium on Computer, Communication, Control and Automation 2010.

APPENDIX

In the table 1, following abbreviations are used.
V: The number of features (the vocabulary size)
N: The number of training documents
Lave = average length of a document
La = Number of tokens
Ld: The average document length (word count)
LV: The average number of unique words in a document
M: The number of training set in categories (M<N)
Ma = types, in the test document
$|D|$ = Number of documents

Table 1. Comparison of classifiers

Classifier Name	Time Complexity	Classifier Principal	Advantages	Disadvantages
KNN	Training $\rightarrow O(NL_d)$ Testing $\rightarrow O(\frac{N}{V} L_v^2) + O(N)$	Distance is computed and K closest samples are selected the category of document is predicted based on the nearest point which has been assigned to particular category Distance is measured $\mathrm{Sim}(Q,D_j) = \frac{\sum_i w_{Q,j} w_{i,j}}{\sqrt{\sum_i w_{Q,j}^2}\sqrt{\sum_i w_{i,j}^2}}$ as	– Effective – Non-parametric – More local characteristics of document are considered comparing with Rocchio	– Classification time is long – Ddifficult to find optimal value of k
RegressionModel (LLSF)[42]	Training time on M categories $\rightarrow O(N^2 K_s)$ Testing time per document -> $O(ML_w)$	The optimization problem in LLSF is : to find W which minimize the sum $\sum_{i=1}^{k}\lVert\vec{e}_i\rVert_2^2 = \sum_{i=1}^{k}\lVert W\vec{a}_i - \vec{b}_i\rVert_2^2 = \lVert WA - B\rVert_F^2$ Where $\vec{e}_i \stackrel{def}{=} W\vec{a}_i - \vec{b}_i$ is mapping error of the i_{th} text pair; the notation $\lVert\cdots\rVert_2$ is vector defined as $\lVert\vec{v}\rVert_2 = \sqrt{\sum_{j=1}^{m} v_j^2}$ And $\vec{v}$ is $m \times 1$; $\lVert\ldots\rVert_F$ is the frobenius matrix norm define as $\lVert M\rVert_F = \sqrt{\sum_{i=1}^{k}\sum_{j=1}^{m} m_{ij}^2}$ and M is m $\times$ k	– it use mean of word instead of matching word	– computational cost higher

Neural Network	Depends upon the selection of learning rate. If the learning rate is too small, then learning will occur at a very slow pace. If the learning rate is too large, then oscillation between inadequate solutions may occur. Thumb rule says: Set learning rate to 1/t, where t is the number of iterations through the training set	$I_j = \sum_i w_{ij} O_i + \theta_j$ Which, computes the net input of unit j with respect to the previous layer, i. $O_j = \frac{1}{1+e^{-I_j}}$ Output of each unit j.	– Produce good results in complex domains – Testing is very fast	– Training is relatively slow – Learned results are difficult for users to interpret than learned rules (comparing with DT)
DNF [29]	-complexity is depend on rule and component of rule ,small number of rule gives less complexity ,large number of rule give more complexity	Rule is constructed in form of 'IF Condition Then Result ' consist disjunctive normal form	– Produce good results in complex domains – Testing is very fast	– Training is relatively slow – Learned results are difficult for users to interpret than learned rules (comparing with DT)
Decision Tree	Training set D -> $O(n \times D \times \log D)$, Where 'n' is the number of attributes describing the tuples in D and	Do the partition of data , D, which is a set of training tuples and their associated class labels; then by making the attribute list, and the set of candidate attributes, Select the attribute by attribute selection methods, a procedure to determine the splitting criterion that gives the 'best' partitions the data tuples into individual classes.	– Easy to understand – Easy to generate rules – Reduce problem complexity	– Training time is relatively expensive – A document is only connected with one branch – Once a mistake is made at a higher level, any sub tree is wrong – Does not handle continuousva riabl – May suffer from over fitting

Rocchio (Linear Classifier)	Training -> $\Theta(|D|L_{ave} + |C||V|$ Testing $\rightarrow \Theta(L_a + |C|M_a) = \Theta(|C|M_a)$ Complexity of computing parameter is Θ ($|C||V|$) since the set of parameters consists of $|C||V|$ conditional probabilities and $|C|$ priors	The average vector over all training document vectors that belong to class c_i, and calculate similarity between test document and each of prototype vectors, which assign test document to the class with maximum similarity. Ci= α * centroid ci – β * centroid ~ci	– Easy to implement – Very fast learner – Efficient in computation	– low classification accuracy – Linear combination too simple for classification – Constant α and β are empirical
Naïve Bayes Classifier	Training $\rightarrow \Theta(|D|L_{ave} + |C||V|)$ Testing $\rightarrow \Theta(L_a + |C||M_a|) = \Theta(|C||M_a|)$ Complexity of computing parameter is Θ $(|C||M_a|)$ since the set of parameters consists of $|C||V|$ conditional probabilities and $|C|$ prior	$C^{*}_{N-1} = \arg\max P(C_j) \prod_{i=1}^{d} |P(w_i|c_j)$ Where, $P(C_j)$= priori probability of class cj ,$P(w_i|c_j)$ = conditional probability of word wi given in cluster cj	– Work well on numeric and textual data – Easy to implement and computation comparing with other algorithms	– Conditional independence assumption is violated by real-world data, perform very poorly when features are highly correlated
Support vector Machines (SVM)	Training time on M categories -> $O(MN^C)$, Testing time per document -> $O(ML_v)$	The optimization of linear SVM is to maximize $\sum_{i=1}^{l} \alpha_i - \frac{1}{2}\sum_{i,j=1}^{l} \alpha_i\alpha_j y_i y_j K(x_i, x_j)$ subject to $\forall i : 0 \leq \alpha_i \leq C$ and $\sum_{i=1}^{l} \alpha_i y_i = 0.$	– Work well on numeric and textual data – Easy to implement and computation	- Conditional independence assumption is violated by real-world data, perform very poorly when features are highly correlated
Associative classifier [58]	Time complexity is addition of time require for mining the rule and time require for rule purring	Three steps process first 3 belong to the training process while the last one represents the classification phase Generate the association rule using association rule mining technique class of new document will assign class depend on which rule is satisfied	-relatively fast at training time -generated rules are easy to understand	-number of terms increase, Increase number of word set-more physical memory require

Voting	It depend on selection of set of classifier and selection of combination function	Idea behind this is consider the result of expert classifiers, and take final result from majority low	– Weak classifier can help to improve accuracy	– Long-time require for result – Accuracy is Dependence an function
Centroid Classifier	If there are 'N' training documents, 'T' test documents, 'W' words in total, K classes and M iteration steps, then complexity to compute the summed centroid and normalized centroid Is *O* (*NW*+ *KW*), since *K*<*N* the time complexity is *O* (*NW*). Overall time complexity of centroid classifier is *O* (*TKW*).	Summed centroid $C_i^S = \sum_{d \in c_i} d$. Normalized Centroid $C_i^N = C_i^S / \lVert C_i^S \rVert_2$ Improved Centroid Classifier [illegible] $C_{A,l}^{**} = \begin{cases} C_{A,l}^{*} & if\ C_{A,l}^{*} \geq 0 \\ 0 & if\ C_{A,l}^{*} < 0 \end{cases}$ $C_A^{***} = C_A^{**} / \lVert C_A^{**} \rVert_2$	– It gives summarize the characteristics of each class	– Sometime training data items that are far away from centre